IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 4, Ver. I (Jul-Aug. 2014), PP 80-86
www.iosrjournals.org

Customized Ontology model for Web Information Gathering using Clustering

[1]Archana Chaugule, [2]Tushar Ghorpade, [3]Puja Padiya
[1]Shah & Anchor Kutchhi Engg.college Mumbai University
[2,3]R.A.I.T. Nerul Mumbai University

Abstract: *The explosion of data leads to the problem on how information should be retrieved accurately and effectively. To address this issue, ontology's are widely used to represent user profiles in personalized web information gathering. As a model for knowledge description and formalization, ontology's are widely used to represent user profiles in personalized web information gathering. When representing user profiles, many models have utilized only knowledge from either a global knowledge base or local knowledge base. Ontology model learns user profiles from both a world knowledge base and local knowledge base. A non-content based customized ontology model is proposed for knowledge representation and reasoning over user profiles. This model generates user Local Instance Repository which includes non-content based descriptors referring to the subjects. This model has improvement over former models in hit/miss ratio, precision and recall parameters.*

Keywords: *ontology model ,customize, user profile, clustering*

I. Introduction

The amount of web-based information available has increased dramatically. How to gather useful information from the web has become a challenging issue for users. Current web information gathering systems attempt to satisfy user requirements by capturing their information needs. For this purpose, user profiles are created for user background knowledge description [2],[4],[8] .User profiles represent the concept models possessed by users when gathering web information A concept model is implicitly possessed by users and is generated from their background knowledge.To simulate user concept models, ontologies—a knowledge description and formalization model—are utilized in personalized web information gathering. Such ontologies are called ontological user profiles [2] or personalized ontologies. To represent user profiles, many researchers have attempted to discover user background knowledge through global or local analysis. Global analysis uses existing global knowledge bases for user background knowledge representation. Commonly used knowledge bases include generic ontologies (e.g., WordNet), thesauruses (e.g., digital libraries) and online knowledge bases (e.g., online categorizations and Wikipedia). The global analysis techniques produce effective Performance for user background knowledge extraction. However, global analysis is limited by the quality of the used knowledge base. For example, Wikipedia was reported as helpful in capturing user interest in some areas but useless for others. Local analysis investigates user local information or observes user behaviour in user profiles. For example, Li and Zhong [8] discovered taxonomical patterns from the users' local text documents to learn ontologies for user profiles. Some groups [2] learned personalized ontologies adaptively from user's browsing history. Alternatively, Sekine and Suzuki analysed query logs to discover user background knowledge.However, because local analysis techniques rely on data mining or classification techniques for knowledge discovery, occasionally the discovered results contain noisy and uncertain information. As a result, local analysis suffers from ineffectiveness at capturing formal user knowledge. From this, we can hypothesize that user background Knowledge can be better discovered and represented if we can integrate global and local analysis within a hybrid model. The goal of ontology learning is to semi-automatically extract relevant concepts and relations from a given corpus or other kinds of data sets to form ontology. A customized ontology model to evaluate this hypothesis is proposed. The ideas which we have implemented are as follows:

1. Global search produces search results based on the existing global knowledge.
2. Local search produces search results based on the user interest which is analysed using user profiles.
3. Content-based clustering is done which searches not only the query with the document name but also with the content present in it.

All local and global repositories have content-based descriptors referring to the subjects. However, a large volume of documents existing on the web may not have such content-based descriptors. To refer that non-content based descriptors clustering technique is used which also groups the documents which does not have descriptors. Compared with other benchmark models customized model is successful.

II. Related Work.

2.1 Ontology Learning

Global knowledge bases were used by many existing models to learn ontologies for web information gathering. For example, Gauch et al. [2] learned personalized ontologies from the Open Directory Project to specify users' preferences and interests in web search. On the basis of the Dewey decimal classification, King et al. developed IntelliOnto to improve performance in distributed web information retrieval. Wikipedia was used by Downey et al. [6] to help understand underlying user interests in queries. These works effectively discovered user background knowledge; however, their performance was limited by the quality of the global knowledge bas Aiming at learning personalized ontologies, many works mined user background knowledge from user local information. Li and Zhong [8] used pattern recognition and association rule mining techniques to discover knowledge from user local documents for ontology construction. . Zhong proposed a domain ontology learning approach that employed various data mining and natural-language understanding techniques. Navigli et al. developed Onto Learn to discover semantic concepts and relations from web documents. Web content mining techniques were used by Jiang and Tan to discover semantic knowledge from domain-specific text documents for ontology learning. Finally, Shehata et al. captured user information needs at the sentence level rather than the document level, and represented user profiles by the Conceptual Ontological Graph. However, the knowledge discovered in these works contained noise and uncertainties.

2.2 User Profiles

User profiles can be categorized into three groups:

1. Interviewing user profiles can be deemed perfect user profiles. They are acquired by using manual techniques, such as questionnaires, interviewing users, and analyzing user classified training sets. One typical example is the TREC Filtering Track training sets, which were generated manually [5]. The users read each document and gave a positive or negative judgment to the document against a given topic. Because, only users perfectly know their interests and preferences, these training documents accurately reflect user background knowledge.
2. Semi-interviewing user profiles are acquired by semi-automated techniques with limited user involvement. These techniques usually provide users with a list of categories and ask users for interesting or non-interesting categories. One typical example is the web training set acquisition model introduced by Tao et al. [3], which extracts training sets from the web based on user feedback categories.
3. Non interviewing techniques do not involve users at all, but ascertain user interests instead. They acquire user profile by observing user activity and behavior and discovering user background knowledge [18]. A typical model is OBIWAN, proposed by Gauch et al. [2], which acquires user profiles based on users' online browsing history .

2.3 World Knowledge Representation

World knowledge is important for information gathering. World knowledge is common-sense knowledge possessed by people and acquired through their experience and education. The world knowledge base must cover an exhaustive range of topics, since users may come from different backgrounds.

1. Broader term- The BT references are for two subjects describing the same topic, but at different levels of abstraction (or specificity). In our model, they are encoded as the " is-a " relations in the world knowledge base.
2. Is-a-Used for many semantic situations, including broadening the semantic extent of a subject and describing compound subjects and subjects subdivided by other topics.
3. Related term- The RT references are for two subjects related in some manner other than by hierarchy. they are encoded as the related-to relations in our world knowledge base.[1].

2.4 Ontology construction

The subjects of user interest are extracted from the WKB via user interaction. A tool called Ontology Learning Environment (OLE) is developed to assist users with such interaction. Regarding a topic, the interesting subjects consist of two sets: positive subjects are the concepts relevant to the information need, and negative subjects are the concepts resolving paradoxical or ambiguous interpretation of the information need. Thus, for a given topic, the OLE provides users with a set of candidates to identify positive and negative subjects. These candidate subjects are extracted from the WKB. In below Fig.1.ontology (partially) constructed for the sample topic "Economic espionage," where the white nodes are positive, the dark nodes are negative, and the grey nodes are neutral subjects. The constructed ontology is customized because the user selects positive and negative subjects for personal preferences and interests. Thus, if a user searches "New York" and plans for a business trip, the user would have different subjects selected and a different ontology constructed, compared to those selected and constructed by a leisure user planning for a holiday.

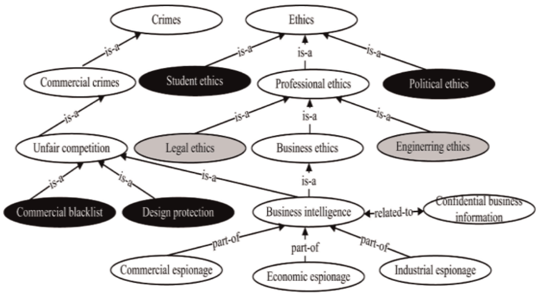

Fig.1: ontology constructed for topic “Economic espionage” [1].

2.5 WordNet

WordNet is a semantic lexicon for the English language. We have utilized it to retrieve the semantic relations between noun keywords of the documents. The benefit of WordNet is that it produces a combination of a dictionary and a thesaurus that is more intuitively usable. It divides the lexicon into five categories: nouns, verbs, adjectives, adverbs, and function words. WordNet group’s words into sets of synonyms called synsets and also provides short, general definitions and examples. Every synset contains a group of synonymous words or collocations; different senses of a word appear in different synsets. Most synsets are connected to other synsets via a number of semantic relationships. These relationships vary based on the type of word. In WordNet nouns are organized in lexical memory as topical hierarchies, verbs are organized by a variety of entailment relations, and adjectives and adverbs are organized as N-dimensional hyperspaces.

2.6 FP Growth algorithm

FP Growth algorithm follows a divide-and-conquer strategy. The original database is compressed or transformed into a tree well known as FP-tree, which holds all the information regarding frequent patterns. The compressed database or FP tree is divided into a set of conditional databases for each frequent item and mines each such database separately to generate frequent patterns. FP-Growth algorithm is an efficient technique for mining frequent patterns from a text document.

2.6.1 Advantages of FP Growth

1. It do not generates large no of candidate items.
2. No repeated scan of original database is required

2.6.2 Algorithm for construction of FP tree:[22].

Input*:* Transaction DB, minimum support threshold.

Output*:* FP-Tree.

```
1. Collect the set of frequent items F and their support.Sort F in support order as prefix.
2. Create the root T of an FP-Tree, and label it as "null". Select and sort F in transaction according to the order of prefix.
3. Let the item list be [p|P], p is the first item and P is remainder for each item list call insertTree(Items, T);
4. function insertTree([p|P], T),if T has child N and N.itemName = p.itemName then N.count++;
        Else
         create node N = p, N.count=1, be linked to T
          ,node- link to the nodes with the  same itemName;
      if P is nonempty then call insertTree(P, N);
```

2.6.3 Algorithm for Mining FP tree:[22].

Input: FP-Tree, minimum support threshold, without DB.
Output: The complete set of frequent patterns.
Method: Call FP-growth (FP-Tree, null)

```
Procedure FP-growth (Tree, α)
{
```

```
if Tree contain a single path P then
for each combination (denote as β) of the nodes in P do
generate pattern βÈα with support = minimum support in β
else for each ai in the Header Table of Tree do {
generate pattern β = aiÈα with support = ai.support
constructβ's conditional pattern base and β's conditional FP-Tree Treeβ;
if Treeβ ≠ null then
call FP-growth (Treeβ, β); } }
```

III. The Proposed Work.

3.1 The Proposed Customized Ontology Model.

Fig. 2 shows the Architecture of proposed Customized Ontology Model . In his two types of search operations are performed.

1. Global search
2. Customized search.

The global search considers the subjects provided in the world knowledge base. The customized search considers only the subjects provided by the individual based on their interests. Clustering is used in the information retrieval systems to enhance the efficiency and effectiveness of the retrieval process. Clustering is a division of data into groups of similar objects. Each group consists of objects that are similar between themselves and dissimilar to objects of the group. In our proposed model the concept of clustering is applied at the initial level i.e. global knowledge representation level, which makes the user to search in the respective domain of the given key word. This will results in effective search and the accurate output.

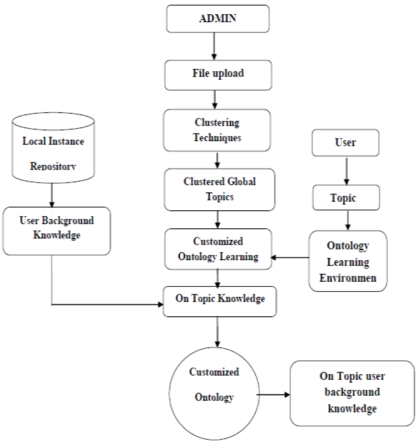
ADMIN
File upload
Clustering Techniques
Local Instance Repository
User
Clustered Global Topics
Topic
User Background Knowledge
Customized Ontology Learning
Ontology Learning Environmen
On Topic Knowledge
Customized Ontology
On Topic user background knowledge

Fig 2. Architecture of the Proposed Customized Ontology Model.

We use relationships between the keywords to cluster the documents. The relationships are retrieved from the Word Net ontology and represented in the form of a graph. The document graphs, which reflect the essence of the documents, are searched in order to find the frequent sub graphs. To discover the frequent sub graphs, we use the Frequent Pattern Growth approach. The common frequent sub graphs discovered by the FP-growth approach are later used to cluster the document. Fig. 3 shows the flow of graph based document clustering using FP growth. The goal of this model is to cluster text and word documents based on their senses rather than keywords, we use Hierarchical Agglomerative Clustering (HAC) technique. HAC for given n elements it creates a hierarchy of clusters such that at the bottom level of the hierarchy every element is considered as a single independent cluster and the top level all the elements are grouped in a single cluster. It does not require more number of clusters as input since the desired number of clusters can be achieved by cutting the hierarchy at a desired level.

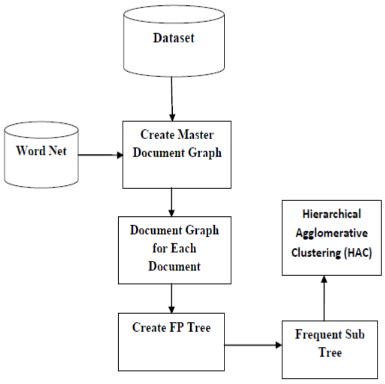
Dataset
Word Net
Create Master Document Graph
Document Graph for Each Document
Hierarchical Agglomerative Clustering (HAC)
Create FP Tree
Frequent Sub Tree

Fig. 3 Flow of graph-based document clustering using FP-growth.

It has two approaches Agglomerative and Divisive. Agglomerative merges the closest pair of elements into a single cluster whereas Divisive groups all the elements in a single cluster. Here we have implemented Group Average method to cluster the documents where the distance between two clusters is defined by the average distance between points in both the clusters and Cosine measure to find the similarity between the clusters. To cluster the documents we have used a dissimilarity matrix which stores the dissimilarity between every pair of document-graphs using the formula dissimilarity = 1-similarity. The value ranges from 0 to 1. User's interest is derived from the analysis of result which he/she searches in the clustered document of the global knowledge repository. The user can perform customized search in which the results for the key word which user inputted is based on both the derived user's interest and the on- topic knowledge. This will result in effective search and produces the accurate output for the user. The global information is retrieved based on the local database which is uploaded by the admin. The uploaded files are clustered because of this time consumption for execution is very less and it gives accurate results, cost is also reduced. The search considers both the content and non-content based descriptors for retrieving the data so it fetches result only when the key word exactly matches with the file name or the content present inside the text document and it produces the absolute results.

3.2 Proposed FP Growth Algorithm

```
Input:  Documents graphs' database DB
        Master Document graph
        Minimum support min_sup
Output: Frequent sub graphs subGraphj
1. Create a list called transactionDB for all DGi € DB
2. Create headerTable for all edge ai € MDG
3. FilterDB (transactionDB,headerTable,min_sup)
4. FPTreeConstructor()
5. FPMining()
6.For each sub graph subGraphi include Subgraph SupportDocs(subGraphj)
```

We proposed this algorithm to discover frequent connected sub graphs. We start by creating a hash table called transactionDB for all the DGs which is similar to original FP-growth procedure. Then headerTable is created from all the edges appearing in the MDG. After that FilterDB() method is called to sort the sub graphs in descending order by frequency based on the min_sup provided by the user. TransactionDB is then updated by pruning the header table at top and bottom for a second time to reduce too specific and abstract edges. After this refinement, FP tree is created by calling the FPTreeConstructor() method. Later, the method FPMining() generates the frequent sub graphs by traversing the FP-tree.

3.3. Algorithm for FP-tree Construction

```
Input: DG database DB and Minimum support threshold, min_sup.
Output: Frequent Pattern Tree (T) made from each DGi in the DB.
1. Scan the DB once.
2. Collect F, the set of edges, and corresponding support of every edge.
3. Sort F in descending order and create FList, the list of frequent edges.
4. Create transactionDB
5. Create the root of an FP-tree T, and label it as "null".
```

```
6. For each DGi in the transactionDB do the following:
7. Select and sort the frequent edges in DGi according to FList.
8. Let the sorted frequent-edge list in DGi be [p|Pi], where p is the first element and Pi is the remaining list.
9. Call insert_tree([p|Pi], T) which performs as follows:
10. If T has a child N such that N.edge_dfs= p.edge_dfs, then N.count++;
            Else  create a new node N, with N.count=1
            Link N to its parent T and link it with the
            same edge_dfs via the node-link structure.
           If Pi is nonempty, call insert_tree([p|Pi], N)
           recursively.
```

3.4. FP-tree Mining Algorithm.

```
Input: The FP-tree T
        Frequent pattern $\alpha$ (at the beginning, $\alpha$ =null)
        Header table hTable, with edges denoted as ai
Output: Frequent patterns $\beta$
1. if T contains a single path P then
2. for each combination (denoted as $\beta$) of the nodes in path P
3. generate pattern ($\beta \cup \alpha$) with support =MIN( supports of all the nodes in $\beta$)
4. else for each ai in the hTtable of T
5. generate pattern $\beta$ = ai$\cup\alpha$ with support = ai.support;
6. construct $\beta$'s conditional pattern base and use it to build $\beta$'s
        conditional FP-tree Tree$\beta$,
        construct $\beta$'s conditional header table hTable$\beta$
7. if Tree$\beta \neq \emptyset$ then
        FP_growth (Tree$\beta$, $\beta$, hTable$\beta$);
```

IV. Conclusion

In proposed work, we have attempted to develop a sense-based clustering mechanism by employing a graph mining technique. Traditional document clustering techniques mostly depend on keywords, whereas our technique relies on the keywords' concepts. The customized ontology model for web information gathering performs better in producing the accurate results by clustering the text documents based on its content. Clustering of documents improves the recall parameter value. This in-turn increases the precision parameter value. Since the correctness of the results is more, the user can find documents relevant to his interest in a single search. This work can be extended in a number of directions. Mining the FP-tree for larger single path lengths is still computationally expensive. We can generate some optimization techniques for computing the combinations of larger single paths which can improve the FP-growth performance.

References

[1]. Xiaohui Tao, Yuefeng Li and NingZhong, "A Personalized Ontology Model for Web Information Gathering", IEEE Transactions on Knowledge and Data Engineering, vol.23, no.4, April 2011.

[2]. S. Gauch, J. Chaffee, and A. Pretschner," Ontology-Based Personalized Search", Web Intelligence and Agent Systems, vol. 1, nos. 3/4 , pp.219-234, 2003.

[3]. X. Tao, Y. Li, N. Zhong, and R. Nayak, "Automatic Acquiring Training Sets for Web Information Gathering," Proc. IEEE/WIC/ACM Int'l Conf. Web Intelligence, pp. 532-535, 2006.

[4]. Y. Li and N. Zhong, "Web Mining Models and Its Applications for Information Gathering" ,Knowledge-Based Systems, vol. 17, pp.207-217, 2004.

[5]. S.E. Robertson and I. Soboroff, "The TREC 2002 Filtering Track Report," proc. Text Retrieval Conf., 2002 .

[6]. D. Downey, S. Dumais, D. Liebling, and E. Horvitz, "Understanding the Relationship between Searchers' Queries and Information Goals," Proc. 17thACM Conf. Information and Knowledge Management (CIKM '08), pp.449-458,2008.

[7] J. Trajkova and S. Gauch, "Improving Ontology-Based User Profiles," Proc. Conf. Research 'Information Assistee par Ordinateur (RIAO '04), pp. 380-389, 2004.

[8]. Kenneth Wai-Ting Leung, Wilfred Ng, and DikLun Lee," Personalized Concept Based Clustering of Search Engine Queries," IEEE Transaction, 2007.

[9]. T. Tran, P. Cimiano, S. Rudolph, and R. Studer," Ontology-Based Interpretation of Keywords for Semantic Search, " Proc. Sixth Int'l Semantic Web and Second Asian Semantic Web Conf.(ISWC '07/ASWC '07), pp.523-536, 2007.

[10]. S.C. punitha, V. Thavavel, M. Punithavalli,"A New Hybrid schemes combining Ontology and clustering for text documents" , ITJ 12,pp. 2447-2453,2013.

[11]. Charanjeet Dadiyala , Prof. Pragati Patil, Prof. Girish Agrawal, " Personalized Web Search ", International Journal of Advanced Research in Computer Science and Software Engineering ", Volume 3, Issue 6, 2013 .

[12]. G.S.Deokate1, Prof. Mrs. V.M.Deshmukh," Personalized Ontology Construction: A Review Study",International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 2, 2013.

[14]. K.V. NarayanaRao, U. Jwalitha, K.D.N.V Rajesh , " Personalized We Information Collection Using Knowledge-based Ontologies", International Journal of Computer Science and Information Technologies, Vol. 3 (5) , pp.5185 - 5189,2012

[15]. Prof. Mrs.V .M. Deshmuk ,Mr.G.S.Deokate," Mining Ontological User ProfilesA Review Study", International Journal Of Computer Science And Applications Vol. 6, No.2, 2013.

[16]. Downey, S. Dumais, D. Liebling, and E. Horvitz, "Understanding the Relationship between Searchers' Queries and Information Goals," Proc. 17th ACM conf. Information and Knowledge Management (CIKM '08), pp. 449-458, 2008.

[17]. Zhicheng Dou, Ruihua Song, Ji-Rong Wen, and Xiaojie Yuan, "Evaluating the Effectiveness of Personalized Web Search",IEEE Transactions on Knowledge and Data Engineering, vol.21, 2009.

[18]. Vincent Schickel Zuber BoiFaltings, " Using Hierarchical Clustering for Learning the Ontologies used in Recommendation Systems" , proceeding on 13th ACM in international conference on knowledge discovery and data mining ., pp. 599-608, 2007.

[19]. Praveen Mukharji, Sowjanyakumar, Harini, " Ontolgy for Automatic Acquisition Web User Information",International Conference on Computer Science and Information Technology.2012.

[20]. R. Agrawal and T. Imilienski Swami. ,"Mining Association Rules between sets of items in large databases.", In Proc. Of ACM SIGMOD, 1993.

[21]. J. Han, J. Pei and Y. Yin, "Mining frequent patterns without Candidate generation. In Proc. ACM-SIGMOD Int. Conf Management of Data(SIGMOD'00), Dallas, pp: 1–12,2000.

[22]. Ahmed A. Mohamed, Sanguthevar Rajasekaran," Query-Based Summarization based on Document Graphs",2006.

[23]. http://www.en.m.wikipideia.org/wiki/wordnet-date:15-01-2014.

[24]. http://www.wordnet.princeton.edu-date:15-01- 2014.

.